J 6

```
MMM         MMM  TTTTTTTTTTTTTTT     AAAAAAAAA        AAAAAAAAA        CCCCCCCCCCCC  PPPPPPPPPPP
MMM         MMM  TTTTTTTTTTTTTTT     AAAAAAAAA        AAAAAAAAA        CCCCCCCCCCCC  PPPPPPPPPPP
MMM         MMM  TTTTTTTTTTTTTTT     AAAAAAAAA        AAAAAAAAA        CCCCCCCCCCCC  PPPPPPPPPPP
MMMMMM   MMMMMM        TTT        AAA         AAA  AAA         AAA  CCC              PPP         PPP
MMMMMM   MMMMMM        TTT        AAA         AAA  AAA         AAA  CCC              PPP         PPP
MMMMMM   MMMMMM        TTT        AAA         AAA  AAA         AAA  CCC              PPP         PPP
MMM   MMM   MMM        TTT        AAA         AAA  AAA         AAA  CCC              PPP         PPP
MMM   MMM   MMM        TTT        AAA         AAA  AAA         AAA  CCC              PPP         PPP
MMM   MMM   MMM        TTT        AAA         AAA  AAA         AAA  CCC              PPP         PPP
MMM         MMM        TTT        AAA         AAA  AAA         AAA  CCC              PPPPPPPPPPPP
MMM         MMM        TTT        AAA         AAA  AAA         AAA  CCC              PPPPPPPPPPPP
MMM         MMM        TTT        AAA         AAA  AAA         AAA  CCC              PPPPPPPPPPPP
MMM         MMM        TTT        AAAAAAAAAAAAAAA  AAAAAAAAAAAAAAA  CCC              PPP
MMM         MMM        TTT        AAAAAAAAAAAAAAA  AAAAAAAAAAAAAAA  CCC              PPP
MMM         MMM        TTT        AAAAAAAAAAAAAAA  AAAAAAAAAAAAAAA  CCC              PPP
MMM         MMM        TTT        AAA         AAA  AAA         AAA  CCC              PPP
MMM         MMM        TTT        AAA         AAA  AAA         AAA  CCC              PPP
MMM         MMM        TTT        AAA         AAA  AAA         AAA  CCC              PPP
MMM         MMM        TTT        AAA         AAA  AAA         AAA     CCCCCCCCCCCC  PPP
MMM         MMM        TTT        AAA         AAA  AAA         AAA     CCCCCCCCCCCC  PPP
MMM         MMM        TTT        AAA         AAA  AAA         AAA     CCCCCCCCCCCC  PPP
```

```
**FILE**ID**MTADEF                                   N  7





MM      MM  TTTTTTTTTT    AAAAAA    DDDDDDDD    EEEEEEEEEE  FFFFFFFFFF
MM      MM  TTTTTTTTTT    AAAAAA    DDDDDDDD    EEEEEEEEEE  FFFFFFFFFF
MMMM  MMMM      TT      AA      AA  DD      DD  EE          FF
MMMM  MMMM      TT      AA      AA  DD      DD  EE          FF
MM  MM  MM      TT      AA      AA  DD      DD  EE          FF
MM  MM  MM      TT      AA      AA  DD      DD  EE          FF
MM      MM      TT      AA      AA  DD      DD  EEEEEEEE    FFFFFFFF
MM      MM      TT      AA      AA  DD      DD  EEEEEEEE    FFFFFFFF
MM      MM      TT      AAAAAAAAAA  DD      DD  EE          FF
MM      MM      TT      AAAAAAAAAA  DD      DD  EE          FF
MM      MM      TT      AA      AA  DD      DD  EE          FF              ....
MM      MM      TT      AA      AA  DD      DD  EE          FF              ....
MM      MM      TT      AA      AA  DDDDDDDD    EEEEEEEEEE  FF              ....
MM      MM      TT      AA      AA  DDDDDDDD    EEEEEEEEEE  FF              ....


BBBBBBBB      333333      222222
BBBBBBBB      333333      222222
BB      BB  33      33  22      22
BB      BB  33      33  22      22
BB      BB          33          22
BB      BB          33          22
BBBBBBBB          33          22
BBBBBBBB          33          22
BB      BB          33      22
BB      BB          33      22
BB      BB  33      33    22
BB      BB  33      33    22
BBBBBBBB      333333    2222222222
BBBBBBBB      333333    2222222222
```

AC
VO

```
!
! Definition file for MTAACP compilation
!
! Version:      'V04-000'
!
!****************************************************************************
!*                                                                          *
!*  COPYRIGHT (c) 1978, 1980, 1982, 1984 BY                                 *
!*  DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.                  *
!*  ALL RIGHTS RESERVED.                                                    *
!*                                                                          *
!*  THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED   *
!*  ONLY IN  ACCORDANCE WITH  THE  TERMS  OF  SUCH  LICENSE  AND WITH THE   *
!*  INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR  ANY  OTHER   *
!*  COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY   *
!*  OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE IS  HEREBY   *
!*  TRANSFERRED.                                                            *
!*                                                                          *
!*  THE INFORMATION IN THIS SOFTWARE IS  SUBJECT TO CHANGE WITHOUT NOTICE   *
!*  AND  SHOULD  NOT  BE  CONSTRUED AS  A COMMITMENT BY DIGITAL EQUIPMENT   *
!*  CORPORATION.                                                            *
!*                                                                          *
!*  DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE  OR  RELIABILITY OF ITS   *
!*  SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.                 *
!*                                                                          *
!*                                                                          *
!****************************************************************************
!
!++
!
! FACILITY: MAGNETIC TAPE ACP
!
! ABSTRACT:
!
!       These are the data stucture definitions and random macros
!       used to compile the MTAACP.
!
! ENVIRONMENT:
!       Starlet Operating System, including privileged system calls
!       and internal system subroutines.
!
!
!--
!
!
! Author:  D. H. Gillespie,      Creation date:  18-may-77  16:25
!
! MODIFIED BY:
!
!       V03-006 MMD0178         Meg Dumont,     26-May-1983  15:14
!               Fix to support new input to IOC$CVT_DEVNAM
!
!       V03-005 MMD0173         Meg Dumont,     9-May-1983  15:15
!               Fix to make USER_STATUS consistently defined within module
!
!       V03-004 MMD0145         Meg Dumont,     25-Apr-1983  18:14
!               Add HDR4 label, add some literals for scratch_offset
!               and file_spec_max
!
!       V03-003 MMD0119         Meg Dumont,     29-Mar-1983  0:44
!               Added misc def's common inside the MTAACP
!
!       V03-002 MMD0002         Meg Dumont,      5-Jan-1983  13:50
!               Add another field to be define for V3.0 systems
!
!       V03-001 MMD0001         Meg Dumont,     11-Nov-1982  10:44
!               Add VCB def for enable user EOT handling. Needed to work on V3.X
!
!       V02-008 DMW00075        David Michael Walp       8-Feb-1982
!               Added VVP${T|S}_ACCOUNT and changed _PROC_NAME to _USERNAME
!
!       V02-007 DMW00058        David Michael Walp       7-Dec-1981
!               Removed MAX_FILESTR_LEN
!
!       V02-006 DMW00047        David Michael Walp      30-Jul-1981
!               Removed MAX_ATTR_CODE for global ATR$C_MAX_CODE, added
!               assume MACRO
!
!       V02-005 DMW00028        David Michael Walp      30-Jul-1981
!               Remove MOU$V_NORDVOL1  Added MOU$V_CHKIFSPC
!
!       V02-004 DMW00027        David Michael Walp      20-Jul-1981
!               Added True and False
!
!       V02-003 DMW00020        David Michael Walp      26-May-1981
!               Added Work_Area_Sz which was a GLOBAL LITERAL from OPCOM.
!               Inceased MSGSIZE from 20 to 124, new OPCOM message format.
!
!       V02-002 MCN0018         Maria del C. Nasr       24-Jun-1980
!               Fix syntax error in the linkage definitions.
!
!       V02-001 REFORMAT        Maria del C. Nasr       17-Jun-1980
!
!       A0103   MCN0003         Maria del C. Nasr       01-Oct-1979   14:45
!               Add HDR3 definition
!
!       A0102   MCN0002         Maria del C. Nasr       26-Sep-79   16:05
!               Change BUG_CHECK macro to use builtin BUGW.
!
!       A0102   SPR20439        D. H. Gillespie,        21-Nov-1978  13:21
!               add number of tape marks spaced
!
!       A0101   DGH0003         D. H. Gillespie         17-NOV-1978  10:00
!               change ERROR() [CODE] to ERROR [CODE] to be compatible with
!               new BLISS
!
```

```
! define common external registers
!
MACRO COMMON_REG = CURRENT_VCB  = 11 : REF BBLOCK%;
MACRO GLOBAL_REG = CURRENT_VCB  = 11%;

! define commonly used linkage
!
LINKAGE
        COMMON_CALL      = CALL : GLOBAL(GLOBAL_REG),
        NOPRES           = CALL : GLOBAL(GLOBAL_REG)
                                   NOPRESERVE(0,1,2,3,4,5,6,7,8,9,10),
        JSB              = JSB  : GLOBAL (GLOBAL_REG),
        L$CLOSE_FILE     = JSB  : GLOBAL(GLOBAL_REG)
                                   NOTUSED(2,3,4,5,6,7,8,9),
        L$GET_REQ        = JSB  : GLOBAL(GLOBAL_REG)
                                   NOTUSED(5,6,7,8,9,10),
        L$GET_START_HDR = JSB  : GLOBAL(GLOBAL_REG)
                                   NOTUSED(2,3,4,5,6,7,8,9),
        L$GTNEXT_VOL_RE = JSB  : GLOBAL(GLOBAL_REG),
        L$GTNEXT_VOL_WR = JSB  : GLOBAL(GLOBAL_REG)
                                   NOPRESERVE(2,3,4,5,6,7,8,9,10),
        L$ISSUE_IO       = JSB  : GLOBAL(GLOBAL_REG)
                                   NOTUSED(2,3,4,5,6,7,8,9,10),
        L$NEXT_VOL_READ = JSB  : GLOBAL(GLOBAL_REG)
                                   NOPRESERVE(2,3,4,5,6,7,8,9,10),
        L$NEXT_VOL_WRIT = JSB  : GLOBAL(GLOBAL_REG)
                                   NOPRESERVE(2,3,4,5,6,7,8,9,10),
        L$PRINT_NOT_LAB = JSB  : GLOBAL(GLOBAL_REG)
                                   NOTUSED(5,6,7,8,9,10),
        L$PRINT_OPR_MSG = JSB  : GLOBAL(GLOBAL_REG)
                                   NOTUSED(2,3,4,5,6,7,8,9,10),
        L$REPOSITION     = JSB  : GLOBAL(GLOBAL_REG)
                                   NOTUSED(2,3,4,5,6,7,8,9,10),
        L$WRAP_AROUND    = JSB  : GLOBAL(GLOBAL_REG)
                                   NOTUSED(2,3,4,5,6,7,8,9,10),
        L$WRITE_HEADER   = JSB  : GLOBAL(GLOBAL_REG)
                                   NOTUSED(7,8,9,10),
        L$WRITE_TM       = JSB  : GLOBAL(GLOBAL_REG)
                                   NOTUSED(2,3,4,5,6,7,8,9,10),
        L$WRITE_TRAILER = JSB  : GLOBAL(GLOBAL_REG)
                                   NOTUSED(2,3,4,5,6,7,8,9,10),
        L$IOC_CVT_DEVNAM = JSB (REGISTER=0,REGISTER=1,REGISTER=4,REGISTER=5; REGISTER=1) :
                                   PRESERVE (2,3,4,5,6)
                                   NOTUSED (7,8,9,10,11);

! define macro to extract size
!
MACRO   $BYTESIZE(OFFSET,POSITION,WIDTH,SIGN) = WIDTH / 8 %;

! declare psect usage to minimize page breakage.
!
PSECT
        OWN      = $LOCKEDD1$,
        GLOBAL   = $LOCKEDD1$,
        PLIT     = $CODE$ (EXECUTE);
```

```
! Declare VAX built in functions.
!
BUILTIN
        CHMU,           ! change mode to user ( ERR_EXIT )
        INSQUE,         ! insert into queue
        MOVTUC,         ! translate strings and  check for invalid characters
        MTPR,           ! move to privilege register ( SET_IPL )
        REMQUE,         ! remove from queue
        ROT;            ! rotate longword

! Structure declarations used for system defined structures to
! save typing.
!
STRUCTURE
        BBLOCK [O, P, S, E; N] =
            [N]
            (BBLOCK+O)<P,S,E>,

        BBLOCKVECTOR [I, O, P, S, E; N, BS] =
            [N*BS]
            (BBLOCKVECTOR+(O+I*BS))<P,S,E>;
```

```
!
! assorted macros used in fcp code
!

! set processor IPL
!
MACRO   SET_IPL (LEVEL) = MTPR (%REF (LEVEL), PR$_IPL)%;

! Declare code that must be locked into the working set.
!
MACRO   LOCK_CODE       =
                PSECT   CODE    = $LOCKEDC1$,
                        PLIT    = $LOCKEDC1$,
                        OWN     = $LOCKEDD1$,
                        GLOBAL  = $LOCKEDD1$;
                        %;

!
!***** NOTE: The following two macros violate the BLISS language definition
!***** in that they make use of the value of SP while building the arg list.
!***** It is the opinion of the bliss maintainers that this usage is safe
!***** from planned future optimizations.
!
! Macro to call the change mode to kernel system service.
! Macro call format is "KERNEL_CALL (ROUTINE, ARG1, ARG2, ... )".
!
MACRO
        KERNEL_CALL (R) =
            BEGIN
            EXTERNAL ROUTINE SYS$CMKRNL : ADDRESSING_MODE (ABSOLUTE);
            BUILTIN SP;
            SYS$CMKRNL(R, .SP, %LENGTH-1
                        %IF %LENGTH GTR 1 %THEN, %REMAINING %FI)
            END%;

!
! Macro to call the change mode to exec system service.
! Macro call format is "EXEC_CALL (ROUTINE, ARG1, ARG2, ... )".
!
MACRO
        EXEC_CALL (R) =
            BEGIN
            EXTERNAL ROUTINE SYS$CMEXEC : ADDRESSING_MODE (ABSOLUTE);
            BUILTIN SP;
            SYS$CMEXEC(R, .SP, %LENGTH-1
                        %IF %LENGTH GTR 1 %THEN, %REMAINING %FI)
            END%;

! Macro used to signal fatal errors (internal consistency checks).
!
MACRO
        BUG_CHECK (CODE) =
                BEGIN
                BUILTIN BUGW;
                EXTERNAL LITERAL %NAME('BUG$_',CODE);
                BUGW(%NAME('BUG$_',CODE) OR 4);
                END
                %;

! Macro to signal an error status and continue.
!
MACRO
        ERROR [CODE] =
            BEGIN
                EXTERNAL USER_STATUS : VECTOR [2];
                BEGIN
                   MAP USER_STATUS : WORD;
                   USER_STATUS = CODE;
                END;
            END
            %;

! Macro to signal an error status and exit.
! Implemented as a call into a change mode to user instruction followed
! by a RET.
!
MACRO
        ERR_EXIT (CODE) =
                (CHMU(%REF (%IF %NULL (CODE) %THEN 0 %ELSE CODE %FI));)%;

! Macro to generate a string with a descriptor.
!
MACRO
        DESCRIPTOR (STRING) =
                UPLIT (%CHARCOUNT (STRING),
                        UPLIT BYTE (STRING))%;

! Macro to return the number of actual parameters supplied to a routine
! call.
!
MACRO
        ACTUALCOUNT =
                BEGIN
                BUILTIN AP;
                .(.AP)<0,8>
                END%;

! check to see that constants have not changed
! e.g. ASSUME (IRC$C_FIXOVHDSZ + 2, IRC$C_VAROVHDSZ);
!
 MACRO ASSUME (A,B) =
        %IF $BYTEOFFSET(A)  NEQ $BYTEOFFSET(B)
        %THEN %WARN('WARNING CONSTANT HAS CHANGED')
        %FI %;
```

```
!+
! The following structures are for the management of virtual pages in MTAACP.
! There is always one virtual page for each volume set.
! Other pages are neeeded when a request must be blocked for volume switch,
! wait for user label request (not implemented), or wait for rewind.
!-

! this structure describes a frre page block
!
MACRO    FVP$L_FORWARD   = 0,0,32,0%;      ! forward link for next free page
MACRO    FVP$L_BACKWARD  = 4,0,32,0%;      ! backwards link for previous free page
MACRO    FVP$W_SIZE      = 8,0,16,0%;      ! # of bytes in the free page block
MACRO    FVP$B_TYPE      = 10,0,8,0%;      ! type of structure
LITERAL  FVP$K_LENGTH    = 12;             ! length of fixed data

! this structure describes a volume virtual page
!
MACRO    VVP$L_FORWARD   = 0,0,32,0%;
MACRO    VVP$L_BACKWARD  = 4,0,32,0%;
MACRO    VVP$B_TYPE      = 10,0,8,0%;
LITERAL  VVP$K_LENGTH    = 12;             ! length of fixed data

! This structure describes fixed data in the first virtual page belonging
! to the volume.
!
MACRO    VVP$T_HDR1      = 12,0,0,0%;      ! HDR1 label
LITERAL  VVP$S_HDR1      = 80;
MACRO    VVP$T_HDR2      = 92,0,0,0%;      ! HDR2 label
LITERAL  VVP$S_HDR2      = 80;
MACRO    VVP$T_HDR3      = 172,0,0,0%;     ! HDR3 label
LITERAL  VVP$S_HDR3      = 80;
MACRO    VVP$T_HDR4      = 252,0,0,0%;     ! HDR4 label
LITERAL  VVP$S_HDR4      = 80;
MACRO    VVP$T_SCRATCH   = 332,0,0,0%;     ! scratch area
LITERAL  VVP$S_SCRATCH   = 80;
MACRO    VVP$L_STATUS    = 412,0,32,0%;    ! IO status - 64 bits long
MACRO    VVP$L_STALLIOFL = 420,0,32,0%;    ! STALLED I/O FORWARD LINK
MACRO    VVP$L_STALLIOBL = 424,0,32,0%;    ! stalled I/O backward link
MACRO    VVP$L_BLOCKDIF  = 428,0,32,0%;    ! block count difference
                                           !  (processed count-tape count)
MACRO    VVP$T_USERNAME  = 432,0,0,0%;     ! username of volume mounter
MACRO    VVP$S_USERNAME  = 12%;            ! same length as JIB$S_USERNAME
MACRO    VVP$T_ACCOUNT   = 444,0,0,0%;     ! username of volume mounter
MACRO    VVP$S_ACCOUNT   = 8%;             ! same length as JIB$S_ACCOUNT
MACRO    VVP$L_NO_TM     = 452,0,32,0%;    ! number of tape marks spaced

ASSUME ( JIB$S_USERNAME, VVP$S_USERNAME );
ASSUME ( JIB$S_ACCOUNT,  VVP$S_ACCOUNT  );

LITERAL SCRATCH_OFFSET = 4 * VVP$S_HDR1; ! The offset is the no of header labels
                                         ! times the size of the ANSI label
! Random other definitions
!
! MOUNT_VOL flags
!
MACRO MOU$V_REWIND   = 0,0,1,0%;  ! rewind on mount
MACRO MOU$V_LBLCHECK = 0,1,1,0%;  ! check label
MACRO MOU$V_CHKIFSPC = 0,2,1,0%;  ! check label if operator specified
MACRO MOU$V_MOUNTERR = 0,3,1,0%;  ! there was an error, force physical mount

LITERAL
        ! some world famous Boolean values
        !
        TRUE            = 1,
        FALSE           = 0,


        ! these are the structure types
        !
        FVP_TYPE        = 1,
        VVP_TYPE        = 2,


        ! this is the size of a mailbox message from the operator
        !
        WORK_AREA_SZ    = 128,
        MSGSIZE         = 124,          ! 4 = status, 4 = id, operator text,
                                        ! plus %OPCOM line (WORK_AREA_SZ - 4)

        IOEFN           = 1,            ! event flag for I/O
        EFN             = 1,            ! event flag for I/O
        TIMEFN          = 3,            ! event flag for timer wait

        EXEC_MODE       = 1,            ! exec_mode value
        USER_MODE       = 3,            ! user_mode access
        MAX_DEVNAM_LENGTH = 16,         ! Set the maxium length that a devname
                                        ! can be with VMS
        NO_OF_SUPPORT_ANSI_LABELS = 4,  ! Number of supported ANSI labels
        ANSI_LBLSZ      = 80,           ! Size of the ANSI standard label
        FILE_SPEC_MAX   = 79;           ! Maximum file specification length
                                        ! for VMS long file names (39.39)
```

0253 AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY

CALDAYNO
LIS

SYSFAC
DAT

ACPCTR
LIS

ALLOCB
LIS

SYSMSG
LIS

CHKACC
LIS

MTAACP

ACCESS
LIS

MTAAACP
MAP

ACCFL
LIS

CALCTV
LIS

BLOCK
LIS

CLOSFL
LIS

MTADEF
B32

CHKDMO
LIS